

Soluciones Diversas

- [Pantalla de bloqueo Windows 11 por GPO](#)
- [GeolP en Proxmox](#)

Pantalla de bloqueo Windows 11 por GPO Configuración de una Imagen Corporativa para la Pantalla de Bloqueo en Windows 11 Pro mediante Active Directory

Objetivo

Este procedimiento permite configurar una imagen corporativa para la pantalla de bloqueo de todos los equipos Windows 11 Pro unidos al dominio utilizando únicamente **Active Directory Group Policy Preferences**, sin necesidad de Microsoft Intune, Windows Enterprise, scripts de inicio o tareas programadas.

La solución fue validada en:

- Windows Server con Active Directory
 - Windows 11 Pro 25H2
 - Group Policy Preferences (Files + Registry)
-

Arquitectura de la solución

La solución utiliza únicamente dos componentes de la GPO:

- **Files**
 - Copia automáticamente la imagen desde NETLOGON hacia cada equipo.
- **Registry**
 - Configura el proveedor de personalización (PersonalizationCSP) utilizado por Windows 11 para establecer la imagen de bloqueo.

No utiliza:

- Startup Scripts
- Logon Scripts
- Scheduled Tasks
- PowerShell
- Intune
- Windows Enterprise

Requisitos

- Active Directory funcional.
- Equipos unidos al dominio.
- Windows 11 Pro 25H2.
- Permisos para modificar Group Policy.
- Imagen corporativa en formato PNG o JPG.

Paso 1. Crear la carpeta compartida

En el controlador de dominio crear la siguiente carpeta dentro de NETLOGON.

```
\\alfiolab.local\NETLOGON\LockScreen
```

Copiar dentro la imagen corporativa.

ALCE-LockScreen.png

La estructura quedará de la siguiente manera.

```
\\alfiolab.local
├── NETLOGON
│   └── LockScreen
│       └── ALCE-LockScreen.png
```

Paso 2. Crear la GPO

Abrir:

Group Policy Management

Crear una nueva GPO.

Nombre recomendado:

LockScreen Corporativo

Vincular la GPO a la OU donde se encuentran los equipos.

Paso 3. Copiar automáticamente la imagen

Ir a:

```
Computer Configuration
├── Preferences
│   └── Windows Settings
│       └── Files
```

Crear un nuevo **File**.

Configuración

Action

Replace

Source file

\\alfiolab.local\NETLOGON\LockScreen\ALCE-LockScreen.png

Destination file

C:\Windows\Web\Screen\ALCE-LockScreen.png

Guardar la configuración.

Con esta preferencia la imagen será copiada automáticamente a cada estación de trabajo.

Paso 4. Configurar el Registro

Ir a:

Computer Configuration

Preferences

Windows Settings

Registry

Crear los siguientes cuatro elementos.

Registry Item 1

Action

Update

Hive

HKEY_LOCAL_MACHINE

Key Path

SOFTWARE\Microsoft\Windows\CurrentVersion\PersonalizationCSP

Value Name

LockScreenImagePath

Value Type

REG_SZ

Value Data

C:\Windows\Web\Screen\ALCE-LockScreen.png

Registry Item 2

Action

Update

Hive

HKEY_LOCAL_MACHINE

Key Path

SOFTWARE\Microsoft\Windows\CurrentVersion\PersonalizationCSP

Value Name

LockScreenImageUrl

Value Type

REG_SZ

Value Data

C:\Windows\Web\Screen\ALCE-LockScreen.png

Registry Item 3

Action

Update

Hive

HKEY_LOCAL_MACHINE

Key Path

SOFTWARE\Microsoft\Windows\CurrentVersion\PersonalizationCSP

Value Name

LockScreenImageStatus

Value Type

REG_DWORD

Base

Decimal

Value Data

1

Registry Item 4

Action

Update

Hive

HKEY_LOCAL_MACHINE

Key Path

SOFTWARE\Policies\Microsoft\Windows\CloudContent

Value Name

DisableWindowsSpotlightFeatures

Value Type

REG_DWORD

Base

Decimal

Value Data

1

Paso 5. Actualizar la política

En una estación ejecutar:

```
gpupdate /force
```

No es necesario reiniciar el equipo.

Paso 6. Verificar la copia de la imagen

Comprobar que exista el siguiente archivo.

```
C:\Windows\Web\Screen\ALCE-LockScreen.png
```

Si el archivo existe, la preferencia **Files** se aplicó correctamente.

Paso 7. Verificar el Registro

Ejecutar:

```
reg query "HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\PersonalizationCSP"
```

Debe mostrar los siguientes valores.

```
LockScreenImagePath
```

```
LockScreenImageUrl
```

```
LockScreenImageStatus
```

Posteriormente verificar:

```
reg query "HKLM\SOFTWARE\Policies\Microsoft\Windows\CloudContent"
```

Debe existir el siguiente valor.

DisableWindowsSpotlightFeatures

Paso 8. Validar la aplicación de la GPO

Ejecutar:

```
gpresult /r /scope computer
```

Debe aparecer la GPO.

LockScreen Corporativo

Paso 9. Probar la pantalla de bloqueo

Presionar:

Windows + L

La pantalla de bloqueo debe mostrar la imagen corporativa configurada.

Actualización de la imagen corporativa

Cuando sea necesario cambiar la imagen únicamente será necesario reemplazar el archivo ubicado en:

```
\\alfiolab.local\NETLOGON\LockScreen\ALCE-LockScreen.png
```

Manteniendo exactamente el mismo nombre.

En la siguiente actualización de políticas:

```
gpupdate /force
```

o durante el procesamiento automático de Group Policy, la nueva imagen será distribuida automáticamente a todos los equipos del dominio.

Validación realizada

Esta solución fue probada satisfactoriamente en múltiples equipos Windows 11 Pro 25H2 unidos al dominio.

Se confirmó el funcionamiento de:

- Copia automática de la imagen mediante Group Policy Preferences.
 - Configuración automática del registro mediante Group Policy Preferences.
 - Aplicación inmediata del nuevo fondo de bloqueo.
 - Actualización automática al modificar la imagen almacenada en NETLOGON.
-

Solución de problemas

La imagen no se copia

Verificar que el archivo exista en:

```
\\alfiolab.local\NETLOGON\LockScreen\ALCE-LockScreen.png
```

y que el equipo tenga permisos de lectura sobre NETLOGON.

La pantalla continúa mostrando imágenes aleatorias

Verificar la existencia de los valores:

```
HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\PersonalizationCSP
```

y

```
HKLM\SOFTWARE\Policies\Microsoft\Windows\CloudContent
```

La GPO no aplica

Ejecutar:

```
gpresult /r /scope computer
```

y confirmar que la GPO **LockScreen Corporativo** aparezca entre las políticas aplicadas.

Conclusión

En Windows 11 Pro 25H2 la política clásica **LockScreenImage** ya no resulta suficiente para forzar una imagen corporativa de bloqueo.

La implementación mediante **Group Policy Preferences (Files + Registry)** utilizando **PersonalizationCSP** constituye una solución sencilla, estable y totalmente administrable desde Active Directory, sin requerir Microsoft Intune, Windows Enterprise, scripts o tareas programadas.

GeoIP en Proxmox

Implementación de GeoIP en Proxmox VE 9 usando IPSet y Firewall

Objetivo

En esta práctica configuraremos **restricción geográfica de acceso (GeoIP) en Proxmox VE 9**, utilizando el firewall integrado de Proxmox y sus **IPSet**.

El objetivo inicial será permitir acceso administrativo a los puertos:

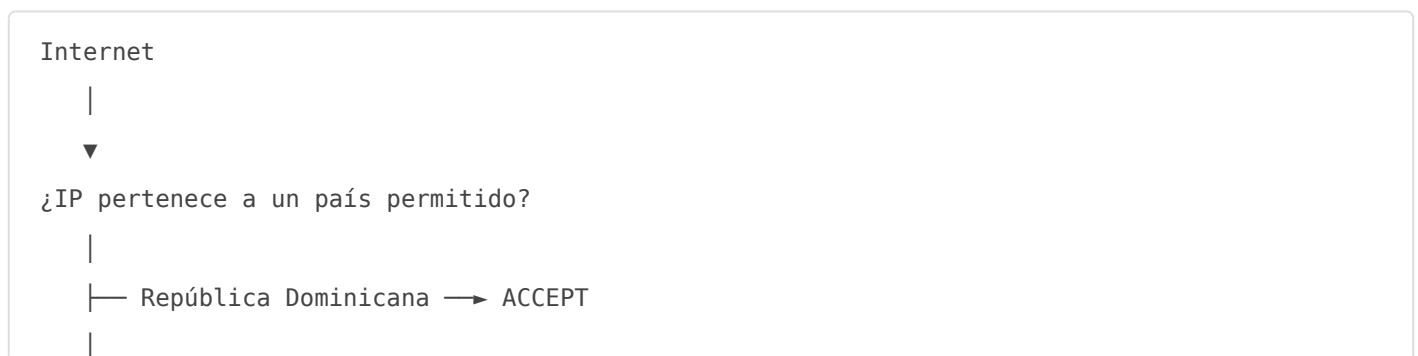
- **TCP 22:** SSH
- **TCP 8006:** interfaz web de Proxmox

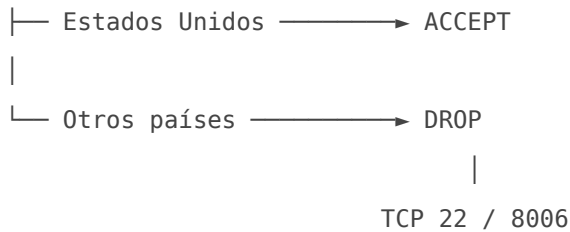
únicamente desde:

- 🇩🇴 **República Dominicana**
- 🇺🇸 **Estados Unidos**

El resto de los países serán bloqueados para esos dos puertos.

La configuración quedará administrable desde la interfaz gráfica de Proxmox:





Además, dejaremos preparado el mecanismo para agregar posteriormente países como Canadá, México, Puerto Rico, España, etc.

1. Requisitos

Esta práctica está diseñada para:

- Proxmox VE 9.x
- Firewall integrado de Proxmox
- Acceso root por SSH
- Acceso a Internet desde el host

Podemos verificar la versión instalada con:

```
pveversion -v | head -20
```

Ejemplo:

```
pve-manager: 9.2.3
proxmox-kernel-helper: 9.2.0
proxmox-kernel-7.0: 7.0.12-1
```

2. Verificar el servicio de firewall

Ejecutar:

```
systemctl status proxmox-firewall --no-pager
```

En Proxmox VE 9 podemos encontrar el servicio:

```
proxmox-firewall.service
```

No debemos confundir que el servicio esté ejecutándose con que las políticas del firewall estén habilitadas.

3. Identificar la red del Proxmox

Ejecutar:

```
ip -br addr
```

y:

```
ip route
```

Debemos identificar:

- IP de administración.
- Interfaz o bridge conectado a Internet.
- Gateway.
- Redes internas.

Ejemplo:

```
vmbr0    66.x.x.x/24
vmbr1    192.168.1.253/24
```

En este ejemplo:

```
vmbr0 = Internet
vmbr1 = Red administrativa interna
```

4. Crear IPSet para República Dominicana

Desde la interfaz web:

Datacenter → Firewall → IPSet → Add

Crear:

Name: geo_do
Comment: Republica Dominicana

No agregar las redes manualmente.

El IPSet será poblado posteriormente mediante un script.

5. Crear IPSet para Estados Unidos

Nuevamente:

Datacenter → Firewall → IPSet → Add

Crear:

Name: geo_us
Comment: Estados Unidos

Ahora tendremos:

```
Datacenter
├── Firewall
│   ├── IPSet
│       └── geo_do
```

6. Crear IPSet para la red administrativa local

Es recomendable permitir siempre el acceso desde nuestra red administrativa interna.

Crear:

Name: management
Comment: Management LAN

Agregar la red correspondiente.

Ejemplo:

192.168.1.0/24

La estructura será:

```
IPSet
└─ management
  └─ 192.168.1.0/24
└─ geo_do
  └─ Redes de República Dominicana
└─ geo_us
  └─ Redes de Estados Unidos
```

7. Verificar la configuración creada por Proxmox

Desde SSH:

```
cat /etc/pve/firewall/cluster.fw
```

Aunque el servidor Proxmox no pertenezca a un cluster, la configuración de firewall a nivel Datacenter se encuentra en:

```
/etc/pve/firewall/cluster.fw
```

Inicialmente veremos algo similar a:

```
[IPSET geo_do] # Republica Dominicana
```

```
[IPSET geo_us] # Estados Unidos
```

```
[IPSET management] # Management LAN
```

```
192.168.1.0/24 # LAN
```

8. Fuente de las redes GeolP

Utilizaremos las listas de redes IPv4 por país publicadas por IPdeny.

El formato es:

```
https://www.ipdeny.com/ipblocks/data/aggregated/XX-aggregated.zone
```

Donde `XX` corresponde al código ISO de dos letras del país.

Ejemplos:

```
do = República Dominicana
us = Estados Unidos
ca = Canadá
mx = México
pr = Puerto Rico
es = España
```

República Dominicana:

```
https://www.ipdeny.com/ipblocks/data/aggregated/do-aggregated.zone
```

Estados Unidos:

```
https://www.ipdeny.com/ipblocks/data/aggregated/us-aggregated.zone
```

9. Crear el script de actualización GeoIP

Crearemos:

```
vi /usr/local/sbin/update-pve-geoip
```

Agregar:

```
#!/bin/bash

set -euo pipefail

FW="/etc/pve/firewall/cluster.fw"
BASE_URL="https://www.ipdeny.com/ipblocks/data/aggregated"

# -----
# PAISES QUE DESEAMOS UTILIZAR
# -----

COUNTRIES="do us"
```

```

# -----

BACKUP="/root/cluster.fw.${date +%Y%m%d-%H%M%S}.bak"

echo "====="
echo " Proxmox GeoIP Updater"
echo "====="
echo

echo "[1] Creando backup..."

cp "$FW" "$BACKUP"

echo "Backup creado:"
echo "$BACKUP"
echo

for COUNTRY in $COUNTRIES
do

    IPSET="geo_${COUNTRY}"
    TMP="/tmp/${COUNTRY}-aggregated.zone"
    URL="${BASE_URL}/${COUNTRY}-aggregated.zone"

    echo "-----"
    echo "Procesando país: $COUNTRY"
    echo "IPSet: $IPSET"
    echo "-----"

    echo "Descargando:"
    echo "$URL"

    curl -fsSL "$URL" -o "$TMP"

    if [ ! -s "$TMP" ]; then
        echo "ERROR: La lista de $COUNTRY está vacía."
        exit 1
    fi

```

```
echo "Validando CIDR..."
```

```
if grep -Ev '^[0-9]+\.[0-9]+\.[0-9]+\.[0-9]+/[0-9]+$' "$TMP"; then
    echo "ERROR: Se encontraron líneas inválidas."
    exit 1
fi
```

```
COUNT=$(wc -l < "$TMP")
```

```
echo "Redes encontradas: $COUNT"
```

```
if [ "$COUNT" -lt 10 ]; then
    echo "ERROR: Cantidad de redes sospechosamente baja."
    exit 1
fi
```

```
python3 - "$FW" "$TMP" "$IPSET" "$COUNTRY" <<'PY'
```

```
import sys
```

```
from pathlib import Path
```

```
fw_file = Path(sys.argv[1])
zone_file = Path(sys.argv[2])
ipset = sys.argv[3]
country = sys.argv[4].upper()
```

```
text = fw_file.read_text()
```

```
start_marker = f"[IPSET {ipset}]"
```

```
start = text.find(start_marker)
```

```
if start == -1:
    raise SystemExit(
        f"ERROR: No existe {start_marker}. "
        "Cree primero el IPSet desde la GUI de Proxmox."
    )
```

```
next_section = text.find("\n[", start + len(start_marker))
```

```

if next_section == -1:
    suffix = ""
else:
    suffix = text[next_section + 1:]

prefix = text[:start]

header_end = text.find("\n", start)

if header_end == -1:
    header = text[start:].rstrip()
else:
    header = text[start:header_end].rstrip()

networks = [
    line.strip()
    for line in zone_file.read_text().splitlines()
    if line.strip()
]

new_section = header + "\n\n"

for network in networks:
    new_section += f"{network} # {country}\n"

new_section += "\n"

if suffix:
    new_text = prefix + new_section + suffix
else:
    new_text = prefix + new_section

fw_file.write_text(new_text)

PY

    echo "$IPSET actualizado correctamente."
    echo

done

```

```
echo "===== "  
echo " ACTUALIZACION COMPLETADA "  
echo "===== "
```

Guardar el archivo.

10. Dar permisos al script

Ejecutar:

```
chmod 700 /usr/local/sbin/update-pve-geoip
```

Verificar:

```
ls -lh /usr/local/sbin/update-pve-geoip
```

11. Ejecutar la actualización

Ejecutar:

```
/usr/local/sbin/update-pve-geoip
```

El resultado será similar a:

```
=====   
Proxmox GeoIP Updater   
=====   
  
[1] Creando backup...   
  
Procesando país: do   
IPSet: geo_do   
  
Redes encontradas: XXX
```

```
geo_do actualizado correctamente.
```

```
Procesando país: us
```

```
IPSet: geo_us
```

```
Redes encontradas: XXXX
```

```
geo_us actualizado correctamente.
```

```
=====
ACTUALIZACION COMPLETADA
=====
```

12. Comprobar los IPSet desde Proxmox

Ir a:

Datacenter → Firewall → IPSet

Abrir:

Ahora aparecerán las redes correspondientes a República Dominicana.

Abrir:

Aparecerán las redes correspondientes a Estados Unidos.

13. Verificar desde consola

Para República Dominicana:

```
grep -A10 '^\[IPSET geo_do\]' /etc/pve/firewall/cluster.fw
```

Para Estados Unidos:

```
grep -A10 '^\[IPSET geo_us\]' /etc/pve/firewall/cluster.fw
```

Ejemplo:

```
[IPSET geo_do] # Republica Dominicana

103.207.44.0/22 # D0
103.7.135.0/24 # D0
...
```

14. Crear las reglas del firewall

Ahora configuraremos el acceso a:

```
TCP 22
TCP 8006
```

Desde:

```
República Dominicana
Estados Unidos
LAN administrativa
```

Ir a:

Datacenter → Firewall → Rules

Crear las siguientes reglas **en este orden**:

Orden	Acción	Source	Protocolo	Puerto
1	ACCEPT	geo_do	TCP	22

Orden	Acción	Source	Protocolo	Puerto
2	ACCEPT	geo_do	TCP	8006
3	ACCEPT	geo_us	TCP	22
4	ACCEPT	geo_us	TCP	8006
5	ACCEPT	management	TCP	22
6	ACCEPT	management	TCP	8006
7	DROP	vacío	TCP	22
8	DROP	vacío	TCP	8006

Todas deben ser:

Direction: IN
Enable: Yes

15. El orden de las reglas es crítico

El firewall procesa las reglas siguiendo su orden.

Por eso:

ACCEPT República Dominicana
ACCEPT Estados Unidos
ACCEPT Management
↓
DROP resto del mundo

Los **DROP** deben estar **después** de los **ACCEPT**.

Una configuración incorrecta como:

DROP TCP 8006
ACCEPT geo_do TCP 8006

bloquearía la conexión antes de alcanzar la regla GeoIP.

16. Verificar las reglas desde consola

Ejecutar:

```
grep -A20 '^\[RULES\]' /etc/pve/firewall/cluster.fw
```

Debemos encontrar algo conceptualmente equivalente a:

```
[RULES]

IN ACCEPT -source +dc/geo_do -p tcp -dport 22
IN ACCEPT -source +dc/geo_do -p tcp -dport 8006

IN ACCEPT -source +dc/geo_us -p tcp -dport 22
IN ACCEPT -source +dc/geo_us -p tcp -dport 8006

IN ACCEPT -source +dc/management -p tcp -dport 22
IN ACCEPT -source +dc/management -p tcp -dport 8006

IN DROP -p tcp -dport 22
IN DROP -p tcp -dport 8006
```

17. Configuración del firewall del nodo

Ir a:

Node → Firewall → Options

Configurar:

Firewall: Yes

Para esta práctica dejamos:

nftables (tech preview): No

No necesitamos habilitar el backend marcado como *tech preview* para utilizar los IPSet y las reglas GeolP de esta práctica.

18. Configuración inicial segura del Datacenter

Ir a:

Datacenter → Firewall → Options

Para nuestra primera implementación utilizaremos:

```
Firewall:      Yes
Input Policy:  ACCEPT
Output Policy: ACCEPT
Forward Policy: ACCEPT
```

¿Por qué Input Policy = ACCEPT?

Porque en esta práctica queremos bloquear específicamente:

```
22
8006
```

mediante nuestras reglas.

Las reglas explícitas:

```
DROP TCP 22
DROP TCP 8006
```

bloquearán esos servicios para países no autorizados.

No estamos implementando todavía una política general:

```
INPUT DROP
```

Esto reduce considerablemente el riesgo de bloquear accidentalmente otros servicios necesarios.

19. Prueba antes de cerrar SSH

Nunca cerrar la sesión SSH utilizada para configurar el firewall hasta comprobar una segunda conexión.

Desde la sesión actual podemos determinar nuestra IP:

```
echo $SSH_CLIENT
```

Ejemplo:

```
190.80.231.94 56569 22
```

La primera dirección es nuestra IP pública:

```
190.80.231.94
```

20. Comprobar que nuestra IP pertenece al país permitido

Podemos comprobar automáticamente si nuestra IP pertenece a `geo_do`.

Ejemplo:

```
python3 - <<'PY'
import ipaddress

ip = ipaddress.ip_address("190.80.231.94")
fw = "/etc/pve/firewall/cluster.fw"

found = False
inside = False

with open(fw) as f:
    for line in f:
        line = line.strip()

        if line.startswith("[IPSET geo_do]"):
            inside = True
            continue

        if inside and line.startswith("["):
            break

        if inside and line and not line.startswith("#"):
            cidr = line.split()[0]

            try:
                network = ipaddress.ip_network(cidr)

                if ip in network:
                    print("IP:", ip)
                    print("MATCH:", network)
                    print("RESULTADO: PERMITIDA")
                    found = True
                    break

            except ValueError:
                pass
```

```
if not found:
    print("IP:", ip)
    print("RESULTADO: NO PERMITIDA")
PY
```

Un resultado correcto sería:

```
IP: 190.80.231.94
MATCH: 190.80.128.0/17
RESULTADO: PERMITIDA
```

Si aparece **NO PERMITIDA**, no activar el firewall hasta investigar la causa.

21. Probar SSH

Sin cerrar la conexión SSH original, abrir otra terminal:

```
ssh root@IP-PROXMOX
```

La nueva conexión debe funcionar desde República Dominicana o Estados Unidos.

22. Probar la interfaz web

Abrir:

```
https://IP-PROXMOX:8006
```

Debe funcionar desde un país permitido.

23. Probar desde un país no autorizado

Podemos utilizar una VPN y seleccionar un país que no esté autorizado.

Por ejemplo:

Canadá
España
Alemania
Reino Unido

Después intentar:

SSH → TCP 22
Web → TCP 8006

El resultado esperado es:

```
PAÍS NO AUTORIZADO
  |
  ▼
No coincide con geo_do
No coincide con geo_us
  |
  ▼
  DROP
 /   \
/     \
TCP 22 TCP 8006
```

24. Agregar otro país

Esta implementación está diseñada para poder ampliarse fácilmente.

Supongamos que queremos agregar **Canadá**.

El código ISO es:

ca

Paso A — Crear IPSet

Desde:

Datacenter → Firewall → IPSet

Crear:

geo_ca

Paso B — Modificar el script

Actualmente tenemos:

COUNTRIES="do us"

Cambiarlo por:

COUNTRIES="do us ca"

Paso C — Ejecutar

/usr/local/sbin/update-pve-geoip

El script detectará:

geo_ca

y descargará automáticamente:

ca-aggregated.zone

Paso D — Crear reglas

Agregar antes de los **DROP**:

ACCEPT +geo_ca TCP 22
ACCEPT +geo_ca TCP 8006

Ahora tendremos:

```
DO ┘
US ┣→ ACCEPT → 22 / 8006
CA ┘

REST0 → DROP → 22 / 8006
```

25. Ejemplo para México

Código:

```
mx
```

Crear:

```
geo_mx
```

Modificar:

```
COUNTRIES="do us ca mx"
```

Ejecutar:

```
/usr/local/sbin/update-pve-geoip
```

Agregar las reglas correspondientes antes de los `DROP`.

26. Códigos útiles

Algunos códigos ISO frecuentes:

País	Código	IPSet
República Dominicana	do	geo_do
Estados Unidos	us	geo_us

País	Código	IPSet
Canadá	ca	geo_ca
México	mx	geo_mx
Puerto Rico	pr	geo_pr
España	es	geo_es
Colombia	co	geo_co
Panamá	pa	geo_pa

La regla general es:

Código ISO: xx

IPSet:

geo_xx

Archivo:

xx-aggregated.zone

27. Actualización automática semanal

Las asignaciones de direcciones IP pueden cambiar, por lo que las listas GeoIP deben actualizarse periódicamente.

Crearemos un servicio systemd:

```
vi /etc/systemd/system/pve-geoip-update.service
```

Agregar:

```
[Unit]
Description=Update Proxmox GeoIP IPSet

[Service]
Type=oneshot
```

```
ExecStart=/usr/local/sbin/update-pve-geoip
```

Ahora crear el timer:

```
vi /etc/systemd/system/pve-geoip-update.timer
```

Agregar:

```
[Unit]
Description=Weekly Proxmox GeoIP Update

[Timer]
OnCalendar=weekly
Persistent=true

[Install]
WantedBy=timers.target
```

Recargar systemd:

```
systemctl daemon-reload
```

Activar:

```
systemctl enable --now pve-geoip-update.timer
```

28. Verificar el timer

Ejecutar:

```
systemctl status pve-geoip-update.timer
```

También:

```
systemctl list-timers | grep geoip
```

Podemos comprobar cuándo será la próxima ejecución automática.

29. Ejecutar una actualización manual

En cualquier momento:

```
/usr/local/sbin/update-pve-geoip
```

El script:

```
Descarga listas
    ↓
Valida formato
    ↓
Comprueba cantidad de redes
    ↓
Crea backup
    ↓
Actualiza geo_do
    ↓
Actualiza geo_us
    ↓
Proxmox aplica la configuración
```

30. Backups

Antes de modificar `cluster.fw`, el script genera automáticamente una copia:

```
/root/cluster.fw.FECHA-HORA.bak
```

Ejemplo:

```
/root/cluster.fw.20260816-213000.bak
```

Esto permite recuperar rápidamente una configuración anterior.

31. Consideraciones de seguridad

GeoIP **no sustituye los mecanismos de autenticación.**

Una dirección permitida por GeoIP únicamente supera la primera barrera de acceso.

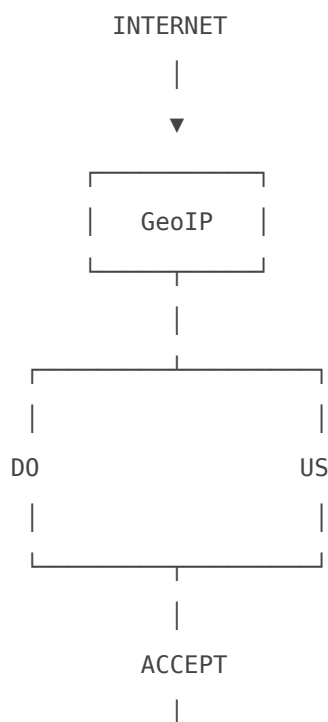
Para SSH se recomienda adicionalmente:

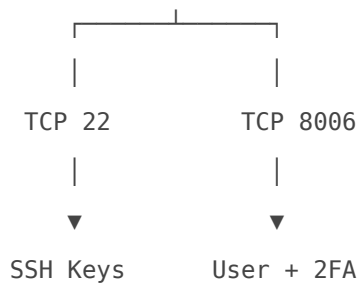
```
Autenticación mediante llave pública
PasswordAuthentication no
PermitRootLogin prohibit-password
```

Para la interfaz de Proxmox:

```
Contraseña robusta
+
2FA
```

La arquitectura de seguridad queda:





Otros países



32. Política INPUT DROP

En esta práctica dejamos:

Input Policy: ACCEPT

y protegemos explícitamente:

22
8006

Es posible endurecer posteriormente el servidor utilizando:

Input Policy: DROP

pero **no debe hacerse simplemente cambiando la opción.**

Antes hay que identificar y permitir explícitamente todos los servicios que el host Proxmox necesite recibir.

Por ejemplo:

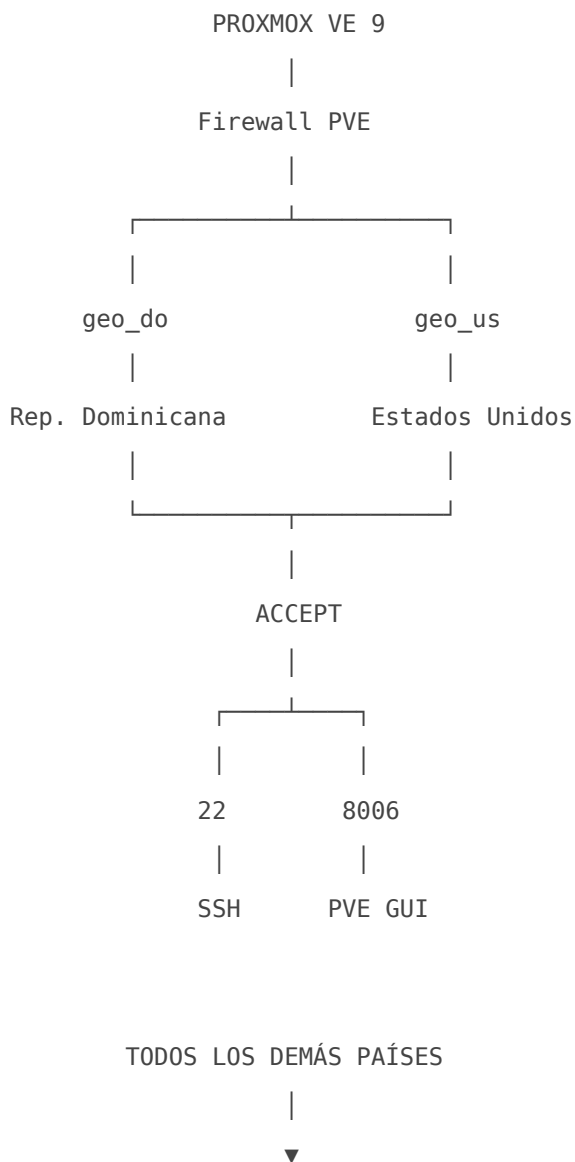
Administración
Cluster
Corosync
Ceph

PBS
Migración
DNS
NTP
Monitoreo
Servicios adicionales

La política **DROP** por defecto debe considerarse una **segunda fase de endurecimiento**.

Resultado final

Al finalizar esta práctica tendremos:



La administración de las reglas permanece disponible desde la **GUI de Proxmox**, mientras que el script mantiene automáticamente actualizadas las redes correspondientes a cada país.

Para agregar países adicionales únicamente debemos:

1. Crear `geo_xx` desde **Datacenter → Firewall → IPSet**.
2. Agregar el código ISO a `COUNTRIES`.
3. Ejecutar `/usr/local/sbin/update-pve-geoip`.
4. Crear los `ACCEPT` correspondientes antes de los `DROP`.
5. Verificar el acceso antes de cerrar la sesión administrativa existente.